Purpose:

The purpose of unit testing is to test individual hardware or software units, or small groups of related units. The emphasis is on removing coding errors (typos, basic logic problems, syntax errors). In some cases, code inspection and walkthroughs are used to verify those units or code paths that are not feasibly tested.

Some contracts do not permit State visibility into unit testing (such as when some of the software is COTS). The text below assumes the State does have visibility into this testing phase.

Assumptions/Pre-Conditions:

The contractor/developer should have performed a code inspection prior to entering unit test, and should have verified that the basic functionality and normal processing paths work correctly prior to beginning unit test.

Expectations:

- Ideally, every code path/line of code that is new or modified should be executed and tested. For paths which are not easily tested, a detailed code inspection should verify functionality. For example, testing of obscure database error handling, such as when mirrored databases fail, is not easily simulated and, unless the probability of such an occurrence is high or the system is "mission critical", these types of scenarios are generally verified through code inspection and inspection of database settings and configuration.
- All possible values should be tested for data entry fields, including errors and unusual entries (such as function key press, control-key combinations, etc.) for all new code and code which has been modified. In some environments, there are tools that will generate test data for this purpose.
- All error cases should be verified and required to end gracefully with the appropriate error data reported (i.e., handle the error; don't allow the program to terminate on an error). This may include executing re-start logic, recovery from the error, or a graceful shutdown.
- All return values should be verified to ensure they are correctly generated under the correct circumstances.
- Screen displays and report formats should be verified for format and data accuracy, including appropriate number of decimal places and correct rounding on calculations, particularly for monetary values.
- New or modified help screens and supporting user materials should be verified.
- Some performance tests may be conducted and used to model or extrapolate behavior.
- Units should "clean up" after themselves, releasing any system resources, as appropriate. Use test tools to check for "memory leaks" and inefficient processing paths.
- If specialized hardware is being used, accuracy and functionality tests may be performed to verify correct interaction of the code and hardware, and that the hardware performs according to its specifications.
- All affected documentation should be updated including in-line code comments and unit/module/function headers, design documents, user manuals, help desk procedures or bulletins, and help files.
- Some error cases or difficult-to-test requirements may be formally verified at this level and/or at the Acceptance test level. A unit-level code inspection may be performed during Acceptance Test to verify non-testable requirements. State staff must be involved in the verification if the inspection is to be considered an "official" verification of the requirement. The Sponsor is encouraged to participate in the verification of requirements at this level, but often declines.

Responsibilities:

- Creation of Tests - Developer
- Execution of Tests - Developer
- Approval of Test Results/Exit Decision (depending on level of State visibility) - Development Manager, Test Manager, QA Manager, Configuration Manager, State Project Manager
- For a complete list of roles and responsibilities, refer to the **Responsibility Assignment Matrix (RAM)** (MS Word)

Environment:

Development Environment

Type of Data:

Artificial data created to follow a particular code path or test specific test cases

Exit Decisions:

- Refer to the **general test exit/acceptance criteria**.

References:

- IEEE Standard **829-1998**, Standard for Software Test Documentation (link to pdf)
- IEEE Standard **1008-1987**, Standard for Software Unit Testing (link to pdf)

Samples:

- None

System Development Contractor Oversight

Responsibility Assignment Matrix

Milestone #4 – Code and Unit Test Completed/Approved

Column 1 lists the expectations for the phase. The remaining columns indicate the expected reviewers (for the Deliverables and Interim Work Products section), or the participants (for the Activities/Decisions and Reviews/Meetings section).

Legend:

P – Primary Responsibility

S – Support Discussions/Activity, as needed

R – Reviewer

A - Approver

I – For Information Only

Note: This matrix assumes that the Prime Contractor has primary responsibility for Unit Testing, and that the project office has some visibility into the process. For M&O projects, the Prime Contractor can be interpreted to be either project or contractor testing staff.

MILESTONE EXPECTATIONS	PROJECT OFFICE MGMT	PROJECT OFFICE CONTRACT MANAGER	PROJECT OFFICE SYSTEMS ENGRING	PROJECT OFFICE QUALITY ASSURANCE	PROJECT OFFICE IMPLMNTN TEAM	PROJECT OFFICE BUSINESS/ INDUSTRY CONSULTANTS	PROJECT OFFICE LEGAL SUPPORT	STAKEHOLDERS / USER REPS	INDEPENDENT VERIFICATION AND VALIDATION	PRIME CONTRACTOR
Deliverables[1]										
Source Code	A		R	R					R	P
Software Development Files (working papers, notes, etc)	A		R	R					R	P

[1] Final versions of deliverables required for exit of this phase.

System Development Contractor Oversight

Responsibility Assignment Matrix

Milestone #4 – Code and Unit Test Completed/Approved

MILESTONE EXPECTATIONS	PROJECT OFFICE MGMT	PROJECT OFFICE CONTRACT MANAGER	PROJECT OFFICE SYSTEMS ENGRING	PROJECT OFFICE QUALITY ASSURANCE	PROJECT OFFICE IMPLMNTN TEAM	PROJECT OFFICE BUSINESS/ INDUSTRY CONSULTANTS	PROJECT OFFICE LEGAL SUPPORT	STAKEHOLDERS / USER REPS	INDEPENDENT VERIFICATION AND VALIDATION	PRIME CONTRACTOR
Unit Test Materials (procedures, scripts, cases, data, etc.)	A		S or R	R					R	P
Unit Test Report	A		S or R	R					R	P
Integration Test Plan	A		S or R	R	R				R	P
BPR Process GAP Analysis	A		R	R	P/R/S			R/S/A	R	P/S
BPR Policy Impact Analysis	A		R	R	P/R/S			R/S/A	R	P/S
BPR Detailed Process Implementation Plan (if not part of the system Implementation Plan)	A		R	R	P/R/S			R/S/A	R	P/S
Updated Workplan	A		R	R	R	R			R	P
Updated Implementation Workplan	A		R	R	R	R			R	P
Updated Data Conversion Workplan	A		R	R	R	R			R	P

System Development Contractor Oversight

Responsibility Assignment Matrix

Milestone #4 – Code and Unit Test Completed/Approved

MILESTONE EXPECTATIONS	PROJECT OFFICE MGMT	PROJECT OFFICE CONTRACT MANAGER	PROJECT OFFICE SYSTEMS ENGRING	PROJECT OFFICE QUALITY ASSURANCE	PROJECT OFFICE IMPLMNTN TEAM	PROJECT OFFICE BUSINESS/ INDUSTRY CONSULTANTS	PROJECT OFFICE LEGAL SUPPORT	STAKEHOLDERS / USER REPS	INDEPENDENT VERIFICATION AND VALIDATION	PRIME CONTRACTOR
Updated Capacity/ Performance Model	A	I	R	R	R	R			R	P
Interim Work Products[2]										
Updated architecture, requirements or design documentation (if applicable)									R	P
Activities/Decisions										
Verify assumptions for the phase are still valid.	A	I	S	S	S	S	S	S	R	P
Validate the Capacity/ Performance Model assumptions and calculations.	A		R	R	I	R		I	R	P

[2] Deliverables which may be in draft form at exit of this phase or which will be expanded in a future phase based on further information (e.g.: preliminary plan vs. final plan).

System Development Contractor Oversight

Responsibility Assignment Matrix

Milestone #4 – Code and Unit Test Completed/Approved

MILESTONE EXPECTATIONS	PROJECT OFFICE MGMT	PROJECT OFFICE CONTRACT MANAGER	PROJECT OFFICE SYSTEMS ENGRING	PROJECT OFFICE QUALITY ASSURANCE	PROJECT OFFICE IMPLMNTN TEAM	PROJECT OFFICE BUSINESS/ INDUSTRY CONSULTANTS	PROJECT OFFICE LEGAL SUPPORT	STAKEHOLDERS / USER REPS	INDEPENDENT VERIFICATION AND VALIDATION	PRIME CONTRACTOR
Verify traceability of the requirements to the code modules, unit tests, and requirements tool.	A		R	R					R	P
Re-validate Deliverable Expectation Documents prior to vendor beginning work on each deliverable.	A	P	P	R	R	R		R	R	R
Perform capacity and throughput tests.	R		R	R	R	R		R	R	P

System Development Contractor Oversight

Responsibility Assignment Matrix

Milestone #4 – Code and Unit Test Completed/Approved

MILESTONE EXPECTATIONS	PROJECT OFFICE MGMT	PROJECT OFFICE CONTRACT MANAGER	PROJECT OFFICE SYSTEMS ENGRING	PROJECT OFFICE QUALITY ASSURANCE	PROJECT OFFICE IMPLMNTN TEAM	PROJECT OFFICE BUSINESS/ INDUSTRY CONSULTANTS	PROJECT OFFICE LEGAL SUPPORT	STAKEHOLDERS / USER REPS	INDEPENDENT VERIFICATION AND VALIDATION	PRIME CONTRACTOR
Verify the department's maintenance and supportability strategies. Ensure the documentation and code are detailed enough to support the strategy.	P	S	S	I	I	S		S	R	S
Review the approach and strategies for integration.	A		R	R	R	R		I	R	P
Review the change control process for requirements and ensure the process is updated to address changes to completed code.	A	S	R	P	R	R	S	R	R	R

System Development Contractor Oversight

Responsibility Assignment Matrix

Milestone #4 – Code and Unit Test Completed/Approved

MILESTONE EXPECTATIONS	PROJECT OFFICE MGMT	PROJECT OFFICE CONTRACT MANAGER	PROJECT OFFICE SYSTEMS ENGRING	PROJECT OFFICE QUALITY ASSURANCE	PROJECT OFFICE IMPLMNTN TEAM	PROJECT OFFICE BUSINESS/ INDUSTRY CONSULTANTS	PROJECT OFFICE LEGAL SUPPORT	STAKEHOLDERS / USER REPS	INDEPENDENT VERIFICATION AND VALIDATION	PRIME CONTRACTOR
Review program/code size and timing budgets for maintainability and adherence to models/-standards.	A	R	R	R	R	R		R	R	P
Reviews/Audits										
Deliverable Review Meetings (see participants for each deliverable listed above)	P						S		R	P
Code Inspections/ Walkthroughs			S	S		I			R	P
Unit Test Inspections/ Walkthroughs			S	S		I			R	P
Integration Test Readiness Review (mini-milestone)	A	I	S	S	I	S	S	S	R	P
QA/CM Audit	A	I	S	S	I				R	P
Phase Closeout Meeting	A	S	S	S	S	S	S	I	R	P

System Development Contractor Oversight

Responsibility Assignment Matrix

Milestone #4 – Code and Unit Test Completed/Approved